Содержание

[Введение 5](#_Toc8940078)

[1 Функциональная организация ЭВМ 6](#_Toc8940079)

[1.1 Связь между функциональной и структурной организацией ЭВМ 6](#_Toc8940080)

[1.2 Обобщенная структура ЭВМ и пути её развития 8](#_Toc8940081)

[1.3 Обрабатывающая подсистема 10](#_Toc8940082)

[1.4 Подсистема памяти 11](#_Toc8940083)

[1.5 Подсистема ввода-вывода 12](#_Toc8940084)

[2 Архитектуры ЭВМ 13](#_Toc8940085)

[2.1 SISD-компьютеры 14](#_Toc8940086)

[2.2 SIMD-компьютеры 15](#_Toc8940087)

[2.3 Матричная архитектура 17](#_Toc8940088)

[2.4 ММХ технология 18](#_Toc8940089)

[2.5 MISD компьютеры 18](#_Toc8940090)

[Заключение 20](#_Toc8940091)

[Список литературы 21](#_Toc8940092)

[Приложение 22](#_Toc8940093)

# Введение

Персональный компьютер (ПК) или персональная ЭВМ (ПЭВМ) — электронная вычислительная машина, с которой может работать пользователь, не являющийся профессиональным программистом.

Она характеризуется развитым (дружественным) человеко-машинным интерфейсом, малыми габаритами, массой, относительно невысокой стоимостью и многофункциональностью (универсальностью) применения.

Одним из основных достоинств ПЭВМ, обеспечивших им потрясающий успех, явился принцип открытой архитектуры, заключающийся в том, что при проектировании ПЭВМ регламентируются и стандартизируются только принцип действия компьютера и его конфигурация (определенная совокупность аппаратных средств и соединений между ними). Построение ПЭВМ не единым неразъемным устройством, а на основе принципа открытой архитектуры (модульности построения) обеспечивает возможность их сборки из отдельных узлов и деталей, разработанных и изготовленных независимыми фирмами-изготовителями. Кроме того, такой компьютер легко расширяется и модернизируется за счет наличия внутренних расширительных разъемов, позволяющих пользователю добавлять разнообразные устройства, удовлетворяющие заданному стандарту, и тем самым устанавливать конфигурацию своей ЭВМ в соответствии со своими личными предпочтениями. Специалисты часто называют такие операции upgrade (расширить, обновить).

Цель данной работы рассмотреть функционально-структурную организацию ЭВМ.

Задачи:

- Рассмотреть состав и назначение основных блоков компьютера.

- Рассмотреть основные характеристики компонентов ЭВМ (центрального процессора и памяти).

# 1 Функциональная организация ЭВМ

* 1. Связь между функциональной и структурной организацией ЭВМ

Существуют два взгляда на построение и функционирование ЭВМ. Первый - взгляд пользователя, не интересующегося технической реализацией ЭВМ и озабоченного только получением некоторого набора функций и услуг, обеспечивающих эффективное решение его задач; второй - разработчика ЭВМ, усилия которого направлены на рациональную техническую реализацию необходимых пользователю функций. С учетом этого обстоятельства и вводятся понятия "функциональная и структурная организация" компьютера.

Функциональная организация ЭВМ - это абстрактная модель ЭВМ, описывающая функциональные возможности машины и предоставляемые ею услуги. Функциональная организация ЭВМ в значительной степени определяется предъявляемыми к ней требованиями, уровнем подготовки потенциальных пользователей, типом решаемых ими задач, потребностями в развитии компьютера (по емкости ЗУ, разрядности, составу периферийных устройств и др.).

Предусматриваемые абстрактной моделью функции ЭВМ реализуются на основе реальных, физических средств (устройств, блоков, узлов, элементов) в рамках определенной структуры. В общем случае под структурной организацией ЭВМпонимается некоторая физическая модель, устанавливающая состав, порядок и принципы взаимодействия основных функциональных частей машины (без излишних деталей их технической реализации).

По степени детальности различают структурные схемы, составленные на уровне

¨ устройств,

¨ блоков,

¨ узлов,

¨ элементов.

Устройство - наиболее крупная функциональная часть ЭВМ, состоящая из элементов, узлов, блоков и выполняющая глобальные операции над кодированными данными (запоминание, обработку, преобразование).

Блок- функциональный компонент ЭВМ, состоящий из элементов и узлов и выполняющий операции над машинными словами или управляющий такими операциями (пример: сумматор, блок регистров).

Узел - часть машины, состоящая из нескольких более простых элементов и представляющая собой сборочную единицу (логическая схема).

Элемент, простейшее устройство ЭВМ, выполняющее одну операцию над входными сигналами. (пример – логический элемент).

Блоки и устройства часто изготавливаются в виде самостоятельных конструктивных модулей.

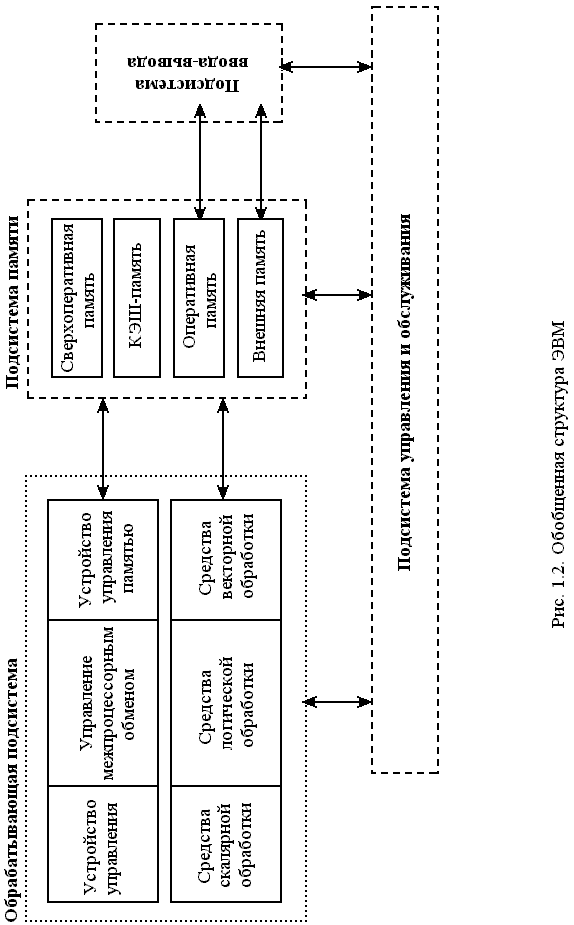
Функциональная организация ЭВМ играет ведущую роль и в значительной степени определяет структурную организацию машины, хотя и не дает жестких ограничений на конечную техническую реализацию структурных элементов. Одна и та же функция может быть реализована на совершенно разных технических средствах.

1.2 Обобщенная структура ЭВМ и пути её развития

Развитие архитектуры неизбежно ведет к развитию структуры ЭВМ. Реализация принципов интеллектуализации, которые все больше определяют развитие архитектуры ЭВМ, возможна при совершенствовании структурной организации, обеспечивающей повышение эффективности вычислительного процесса и, как следствие этого, рост производительности ЭВМ. В конечном счете, условием и критерием развития структуры является рост производительности ЭВМ.

Основной тенденцией в развитии структуры ЭВМ является разделение функций системы и максимальная специализация подсистем для выполнения этих функций.

Обобщенная структура ЭВМ приведена на рис.1.2.

   
Обобщенная структура ЭВМ состоит из следующих составных частей:

1¨ обрабатывающей подсистемы;

1. ¨ подсистемы памяти;
2. ¨ подсистемы ввода-вывода;
3. ¨ подсистемы управления и обслуживания.
4. Для каждой подсистемы выделены основные направления их развития.

## 1.3 Обрабатывающая подсистема

Развитие обрабатывающей подсистемы в большей степени, чем всех остальных подсистем, идет по пути разделения функций и повышения специализации составляющих ее устройств. Создаются специальные средства, которые осуществляют функции управления системой, освобождая от этих функций средства обработки. Такое распределение функций сокращает эффективное время обработки информации и повышает производительность ЭВМ. В то же время средства управления, как и средства обработки, становятся более специализированными. Устройство управления памятью реализует эффективные методы передачи данных между средствами обработки и подсистемой памяти. Меняются функции центрального устройства управления. С одной стороны, ряд функций передается в другие подсистемы (например, функции ввода-вывода), с другой - развиваются средства организации параллельной обработки нескольких команд (суперскалярная обработка) с одновременным повышением темпа исполнения последовательности команд. Для повышения темпа выполнения последовательности команд применяются методы конвейерной обработки наряду с совершенствованием алгоритмов диспетчеризации и исполнения команд. Бурно развивается управление межпроцессорным обменом как эффективное средство передачи информации между несколькими центральными процессорами, входящими в состав вычислительной системы или комплекса.

Операционные устройства (АЛУ) обрабатывающей подсистемы, кроме традиционных средств скалярной (суперскалярной) и логической обработки, все шире стали включать специальные средства векторной обработки. При этом время выполнения операций можно резко сократить как за счет использования арифметического конвейера (одного или нескольких), так и за счет сокращения такта работы конвейера. Возможности задач к распараллеливанию алгоритма счета снимают принципиальные ограничения к организации параллельной обработки информации и использованию структур с глубокой конвейеризацией. В устройствах скалярной обработки все шире появляются специальные операционные блоки, оптимизированные на эффективное выполнение отдельных операций.

## 1.4 Подсистема памяти

Подсистема памяти современных компьютеров имеет иерархическую структуру, состоящую из нескольких уровней:

¨ сверхоперативный уровень (локальная память процессора, кэш-память первого и второго уровня);

¨ оперативный уровень (оперативная память, дисковый кэш);

¨ внешний уровень (внешние ЗУ на дисках, лентах и т.д.).

Каждый уровень состоит из запоминающих устройств, обладающих различным быстродействием и емкостью. Чем выше уровень, тем выше быстродействие соответствующей памяти, но меньше её емкость.

Эффективными методами повышения производительности ЭВМ являются увеличение количества регистров общего назначения процессора, использование многоуровневой кэш-памяти, увеличение объема и пропускной способности оперативной памяти, буферизация передачи информации между ОП и внешней памятью. Увеличение пропускной способности оперативной памяти достигается за счет увеличения их расслоения и секционирования.

* 1. Подсистема ввода-вывода

В состав подсистемы ввода-вывода входит набор специализированных устройств, между которыми распределены функции ввода-вывода, что позволяет свести к минимуму потери производительности системы при операциях ввода-вывода. Эти устройства можно условно разделить на критичные и некритичные по быстродействию. К критичным по быстродействию устройствам относятся обработчики команд ввода-вывода и контроллеры интерфейсов. Эти устройства определяют пропускную способность подсистемы ввода-вывода. Некритичные по быстродействию устройства управляют распределением линий в подсистеме ввода-вывода.

Основными направлениями развития подсистем ввода-вывода являются канальная технология ввода-вывода, матричная топология коммутации периферийных устройств (ПУ), увеличение количества и пропускной способности каналов.

# 2 Архитектуры ЭВМ

Под архитектурой ЭВМ понимается функциональная и структурная организация машины, определяющая методы кодирования данных, состав, назначение, принципы взаимодействия технических средств и программного обеспечения.

Можно выделить следующие важные для пользователя группы характеристик ЭВМ, определяющих её архитектуру:

1) характеристики и состав модулей базовой конфигурации ЭВМ;

2) характеристики машинного языка и системы команд (количество и номенклатура команд, их форматы, системы адресации, наличие программно-доступных регистров в процессоре и т.п.), которые определяют алгоритмические возможности процессора ЭВМ;

3) технические и эксплуатационные характеристики ЭВМ;

4) состав программного обеспечения ЭВМ и принципы его взаимодействия с техническими средствами ЭВМ.

К наиболее общему принципу классификации ЭВМ и систем по типам архитектуры следует отнести разбиение их на однопроцессорные и многопроцессорные архитектуры (см. рис.2.1).

Исторически первыми появились однопроцессорные архитектуры. Классическим примером однопроцессорной архитектуры является архитектура фон Неймана со строго последовательным выполнением команд: процессор по очереди выбирает команды программы и также по очереди обрабатывает данные (программа и данные хранятся в единственной последовательно адресуемой памяти).

По мере развития вычислительной техники архитектура фон Неймана обогатилась сначала конвейером команд, а затем многофункциональной обработкой, и по таксономии\* М.Флина получила обобщенное название компьютера с одним потоком команд и одним потоком данных.

Поток команд - это последовательность команд, выполняемых ЭВМ (системой), а поток данных - последовательность данных (исходная информация и промежуточные результаты решения задачи), обрабатываемых под управлением потока команд.

## 2.1 SISD-компьютеры

## 

https://konspekta.net/infopediasu/baza1/1660140438908.files/image003.png 

Рис. 2.2. SISD- архитектура

SISD (Single Instruction Single Data) или ОКОД - один поток команд, один поток данных. SISD компьютеры это обычные, "традиционные" последовательные компьютеры, в которых в каждый момент времени выполняется лишь одна операция над одним элементом данных (числовым или каким-либо другим значением). При работе такой системы в мультипрограммном режиме, когда совместно решаются несколько задач (программы и исходные данные по каждой из них хранятся в оперативной памяти), обеспечивается параллельная работа устройств системы, происходит разделение времени и оборудования между совместно выполняемыми программами. Но в каждый данный момент операционное устройство (АЛУ), поскольку оно является единственным, занимается обработкой информации по какой-то одной команде, т. е. одновременное преобразование информации в АЛУ по нескольким командам, принадлежащим разным участкам одной и той же программы или разным программам, невозможно. Основная масса современных ЭВМ функционирует в соответствии с принципом фон Неймана и имеет архитектуру класса SISD. Данная архитектура породила CISC, RISC и архитектуру с суперскалярной обработкой.

* 1. SIMD-компьютеры

SIMD (Single Instruction Stream - Multiple Data Stream) или ОКМД - один поток команд и множество потоков данных. SIMD компьютеры состоят из одного командного процессора (управляющего модуля), называемого контроллером, и нескольких модулей обработки данных, называемых процессорными элементами.

|  |
| --- |
|  |
|  | https://konspekta.net/infopediasu/baza1/1660140438908.files/image005.png https://konspekta.net/infopediasu/baza1/1660140438908.files/image006.png |

Рис. 2.3. SIMD- архитектура

Управляющий модуль принимает, анализирует и выполняет команды. Если в команде встречаются данные, контроллер рассылает на все процессорные элементы команду, и эта команда выполняется на нескольких или на всех процессорных элементах.

Все процессорные элементы идентичны и каждый из них представляет собой совокупность управляюще-обрабатывающего органа (быстродействующего процессора) и процессорной памяти небольшой емкости. Процессорные элементы выполняют операции параллельно над разными потоками данных (ПД) под управлением общего потока команд (ПК), вследствие чего такие ЭВМ называются системами с общим потоком команд. В любой момент в каждом процессоре выполняется одна и та же команда, но обрабатываются различные данные. Реализуется синхронный параллельный вычислительный процесс.

Одним из преимуществ данной архитектуры считается то, что в этом случае более эффективно реализована логика вычислений. До половины логических инструкций обычного процессора связано с управлением выполнением машинных команд, а остальная их часть относится к работе с внутренней памятью процессора и выполнению арифметических операций. В SIMD компьютере управление выполняется контроллером, а "арифметика" отдана процессорным элементам. Возможны два способа построения компьютеров этого класса. Это матричная структура ЭВМ и векторно-конвейерная обработка.

2.3 Матричная архитектура

Суть матричной структуры заключается в том, что имеется множество процессорных элементов, исполняющих одну и ту же команду над различными элементами вектора (потоков данных), объединенных коммутатором. Каждый процессорный элемент включает схемы местного управления, операционную часть, схемы связи и собственную оперативную память. Изменение производительности матричной системы достигается за счет изменения числа процессорных элементов.

Основные их преимущества - высокая производительность и экономичность. Недостатки матричных систем, ограничивающие области их применения, заключаются в жесткости синхронного управления матрицей процессорных элементов и сложности программирования обмена данными между процессорными элементами через коммутатор.

Они применяются главным образом для реализации алгоритмов, допускающих параллельную обработку многих потоков данных по одной и той же программе (одномерное и двумерное прямое и обратное преобразования Фурье, решение систем дифференциальных уравнений в частных производных, операций над векторами и матрицами и др.). Матричные системы довольно часто используются совместно с универсальными однопроцессорными ЭВМ. Примером векторных супер-ЭВМ с матричной структурой является знаменитая в свое время система ILLIAC-IV.

* 1. ММХ технология

Еще одним примером SIMD-архитектуры является технология ММХ, которая существенно улучшила архитектуру микропроцессоров фирмы Intel. Технология MMX представляет собой компромиссное решение, объединяющее пути, используемые в классическом процессоре CISC-архитектуры (Pentium), в компьютерах с параллельной SIMD-архитектурой, с добавлением ряда простых (RISC) команд параллельной обработки данных. Она разработана для ускорения выполнения мультимедийных и коммуникационных программ с добавлением новых типов данных и новых инструкций. Технология в полной мере использует параллелизм SIMD-архитектуры и сохраняет полную совместимость с существующими операционными системами и приложениями для SISD.

Точно также, как однопроцессорные компьютеры, представлены архитектурами с одним потоком данных SISD и множеством потоков данных SIMD, так и многопроцессорные системы могут быть представлены двумя базовыми типами архитектур в зависимости от параллелизма данных:

2.5 MISD компьютеры

MISD (Multiple Instruction Stream - Single Data Stream) или МКОД - множество потоков команд и один поток данных. MISD компьютеры представляет собой, как правило, регулярную структуру в виде цепочки последовательно соединенных процессоров П1, П2, ..., ПN, образующих процессорный конвейер (рис. 2.4). В такой системе реализуется принцип конвейерной (магистральной) обработки, который основан на разбиении всего процесса на последовательно выполняемые этапы, причем каждый этап выполняется на отдельном процессоре. Одинарный поток исходных данных для решения задачи поступает на вход процессорного конвейера. Каждый процессор решает свою часть задачи, и результаты решения в качестве исходных данных передает на вход последующего процессора. К каждому процессору подводится свой поток команд, т. е. наблюдается множественный поток команд ПК1, ПК2, ..., ПКN.

Вычислительных машин такого класса практически нет и трудно привести пример их успешной реализации. Один из немногих - систолический массив процессоров, в котором процессоры находятся в узлах регулярной решетки, роль ребер которой играют межпроцессорные соединения. Все процессорные элементы управляются общим тактовым генератором. В каждом цикле работы каждый процессорный элемент получает данные от своих соседей, выполняет одну команду и передает результат соседям.

В дальнейшем для MISD нашлась ещё одна адекватная организация вычислительной системы - распределенная мультипроцессорная система с общими данными. Наиболее простая и самая распространенная система этого класса - обычная локальная сеть персональных компьютеров, работающая с единой базой данных, когда много процессоров обрабатывают один поток данных. Впрочем, тут есть одна тонкость. Как только в такой сети все пользователи переключаются на обработку собственных данных, недоступных для других абонентов сети, MISD - система превращается в систему с множеством потоков команд и множеством потоков данных, соответствующую MIMD-архитектуре.

Заключение

В данной работе была раскрыта структурная схема ЭВМ. Структура компьютера – это некоторая модель, устанавливающая состав, порядок и принципы взаимодействия входящих в нее компонентов. Центральной частью компьютера является системный блок, с присоединенными к нему клавиатурой, монитором и мышью. Описаны основные компоненты ЭВМ: микропроцессор, внешняя и внутренняя память, генератор тактовой частоты, контроллеры (в том числе контроллер прерывания), системная шина, устройства ввода и вывода информации и другие.

Список литературы

1. Акулов О. А., Медведьев Н. В. Информатика: базовый курс. М.: Омега-Л, 2006.

2. Дорот В. А., Новиков Ф. Н. Толковый словарь современной компьютерной лексики. 2-е изд. СПб.: BHV, 2001.

3. Информатика: Учебник. Под ред. Макаровой Н. В. М.: Финансы и статистика, 2000.

4. Лесничая И.Г. Информатика и информационные технологии. Учебное пособие. М.: Издательство Эксмо, 2007

5. Попов В.Б. Основы компьютерных технологий. М. : Финансы и статистика, 2002.

# Приложение

Вариант 2

|  |  |  |
| --- | --- | --- |
| **Задание** | **Используемые ВУ** | **Пояснения** |
| Программа ввода символов с клавиатуры с выводом на дисплей | Клавиатура  Дисплей  Таймер | Очистка буфера клавиатуры после ввода 50 символов или каждые 10 секунд |

Реализация программы:

RDI #101;Команда очистки буфера

OUT 1 ;для клавиатуры

OUT 1;и дисплея

RDI #102;Команда сброса ошибки

OUT 1 ;для клавиатуры

OUT 11 ;и для дисплея

RDI #10 ;Команда разрешения МВУ

OUT 1 ;для клавиатуры без прерываний (по опросу)

OUT 11;для монитора без автоувеличения адреса

RD #100 ;вводим переменную 100

WR 90 ;заносим в ячейку 90

RD #10 ;счётчик

WR 91 ;заносим в 91

RDI #1011 ;вкл 1 таймера

OUT 22

RDI #0 ;сброс таймера

OUT 21

INPUT:

IN 21 ;считать время с 1-го таймера

SBI 1000 ;---- -10 сек

JS NX ; если не прошло то...

RDI #0 ;сброс таймера

OUT 21

RDI #101 ;очистка буфера

OUT 1

NX:

IN 0 ;.получим из буфера код последней клавиши

JZ INPUT ;если буфер пуст, то далее по циклу

OUT 10 ;на экран последнюю клавишу

WR @90 ;записываем код в указанную ячейку

RD 90 ;считываем 100 или больше

ADD #1 ; прибавляем 1 к 100

WR 90 ;записываем в ячейку 90

RD 91 ;счётчик

SUB #1 ;уменьшаем на 1

WR 91

JNZ M1 ;условие если не равно 0, то прыжок на М1

RD #100

WR 90

RD #10

WR 91

M1:

RD R1 ;здесь, получение индекса ячейки

ADD #1 ;увеличим на 1

WR R1 ;сохраним в R1

OUT 13 ;в порт новое значение

RD R0 ;в R0 храним кол-во введ. символов

ADD #1 ;увеличим на 1

WR R0 ;сохраним

SBI 50;проверим, достигло ли оно разрешённого кол-ва(50)

JNZ INPUT ;если нет то по циклу

RD #0;иначе обнулим

WR R0

RDI #101 ;и очистим буфер

OUT 1

JMP INPUT ;по бесконечному циклу